

1 Stable task information from an 2 unstable neural population

3 Michael E. Rule¹, Adrianna R. Loback¹, Dhruva V. Raman¹, Laura N. Driscoll²,
4 Christopher D. Harvey³, Timothy O'Leary¹

*For correspondence:
tso24@cam.ac.uk (TOL)

5 ¹Department of Engineering, University of Cambridge, Cambridge, Cambridgeshire, CB2
6 1PZ, UK; ²Department of Electrical Engineering, Stanford University, Stanford, CA, CA
7 94305-9505, USA; ³Department of Neurobiology, Harvard Medical School, Boston, MA,
8 02115, USA

9

10 **Abstract** Over days and weeks, neural activity representing an animal's position and movement
11 in sensorimotor cortex has been found to continually reconfigure or 'drift' during repeated trials
12 of learned tasks, with no obvious change in behavior. This challenges classical theories which
13 assume stable engrams underlie stable behavior. However, it is not known whether this drift
14 occurs systematically, allowing downstream circuits to extract consistent information. Analyzing
15 long-term calcium imaging recordings from posterior parietal cortex in mice (*Mus musculus*), we
16 show that drift is systematically constrained far above chance, facilitating a linear weighted
17 readout of behavioural variables. However, a significant component of drift continually degrades
18 a fixed readout, implying that drift is not confined to a null coding space. We calculate the amount
19 of plasticity required to compensate drift independently of any learning rule, and find that this is
20 within physiologically achievable bounds. We demonstrate that a simple, biologically plausible
21 local learning rule can achieve these bounds, accurately decoding behavior over many days.

22

23 Introduction and Results

24 A core principle in neuroscience is that behavioral variables are represented in neural activity. Such
25 representations must be maintained to retain learned skills and memories. However, recent work
26 has challenged the idea of long-lasting neural codes (*Rumpel and Triesch, 2016*). In our recent
27 work (*Driscoll et al., 2017*), we found that neural activity-behavior relationships in individual pos-
28 terior parietal cortex (PPC) neurons continually changed over many days during a repeated virtual
29 navigation task. Similar 'representational drift' has been shown in other neocortical areas and
30 hippocampus (*Attardo et al., 2015; Ziv et al., 2013; Levy et al., 2019*). Importantly, these studies
31 showed that representational drift is observed in brain areas essential for performing the task long
32 after the task has been learned.

33 These experimental observations raise the major question of whether drifting representations
34 are fundamentally at odds with the storage of stable memories of behavioral variables (e.g. *Gan-*
35 *guly and Carmena, 2009; Tonegawa et al., 2015*). Theoretical work has proposed that a consistent
36 readout of a representation can be achieved if drift in neural activity patterns occurs in dimensions
37 of population activity that are orthogonal to coding dimensions - in a 'null coding space' (*Rokni et al.,*
38 *2007; Druckmann and Chklovskii, 2012; Ajemian et al., 2013; Singh et al., 2019*). This can be facil-
39 itated by neural representations that consist of low-dimensional dynamics distributed over many
40 neurons (*Montijn et al., 2016; Gallego et al., 2018, 2020; Hennig et al., 2018; Degenhart et al., 2020*).
41 Redundancy could therefore permit substantial reconfiguration of tuning in single cells without dis-
42 rupting neural codes (*Druckmann and Chklovskii, 2012; Huber et al., 2012; Kaufman et al., 2014*;

43 *Ni et al., 2018; Kappel et al., 2018*). However, the extent to which drift is confined in such a null
44 coding space remains an open question.

45 Purely random drift, as would occur if synaptic strengths and other circuit parameters follow
46 independent random walks, would eventually disrupt a population code. Several studies have pro-
47 vided evidence that cortical synaptic weights and synaptic connections exhibit statistics that are
48 consistent with a purely random process (*Moczulska et al., 2013; Loewenstein et al., 2011, 2015*).
49 Indeed, our previous experimental findings reveal that drift includes cells that lose representa-
50 tions of task relevant variables, suggesting that some component of drift affects coding dimensions
51 (*Driscoll et al., 2017*).

52 Together, these observations raise fundamental questions that have not been directly addressed
53 with experimental data, and which we address here. First, to what extent can ongoing drift in task
54 representations be confined to a null coding space over extended periods while maintaining an ac-
55 curate readout of behavioural variables in a biologically plausible way? Second, how might we esti-
56 mate how much additional ongoing plasticity (if any) would be required to maintain a stable read-
57 out of behavioural variables, irrespective of specific learning rules? Third, is such an estimate of
58 ongoing plasticity biologically feasible for typical levels of connectivity, and typical rates of change
59 observed in synaptic strengths? Fourth, can a local, biologically plausible plasticity mechanism tune
60 readout weights to identify a maximally stable coding subspace and compensate any residual drift
61 away from this subspace?

62 We addressed these questions by modelling and analysing data from *Driscoll et al. (2017)*. This
63 dataset consists of optical recordings of calcium activity in populations of hundreds of neurons in
64 Posterior Parietal Cortex (PPC) during repeated trials of a virtual reality T-maze task (1a). Mice were
65 trained to associate a visual cue at the start of the maze with turning left or right at a T-junction.
66 Behavioral performance and kinematic variables were stable over time with some per-session vari-
67 ability (mouse 4 exhibited a slight decrease in forward speed; Fig. 2-S1). Full experimental details
68 can be found in the original study.

69 Previous studies identified planning and choice-based roles for PPC in the T-maze task (*Harvey
70 et al., 2012*), and stable decoding of such binary variables was explored in (*Driscoll et al., 2017*).
71 However, in primates PPC has traditionally been viewed as containing continuous motor-related
72 representations (*Andersen et al., 1997; Andersen and Buneo, 2002; Mulliken et al., 2008*), and re-
73 cent work (*Krumin et al., 2018; Minderer et al., 2019*) has confirmed that PPC has an equally motor-
74 like role in spatial navigation in rodents (*Calton and Taube, 2009*). It is therefore important revisit
75 these data in the context of continuous kinematics encoding.

76 Previous analyses showed that PPC neurons activated at specific locations in the maze on each
77 day. When peak activation is plotted as a function of (linearized) maze location, the recorded pop-
78 ulation tiles the maze, as shown in Figure 1b. However, maintaining the same ordering in the
79 same population of neurons revealed a loss of sequential activity over days to weeks (top row of
80 1b). Nonetheless, a different subset of neurons could always be found to tile the maze in these
81 later experimental sessions. In all cases, the same gradual loss of ordered activation was observed
82 (second and third rows, 1b). Figure 1c shows that PPC neurons gain or lose selectivity and occa-
83 sionally change tuning locations. Together, these data show that PPC neurons form a continually
84 reconfiguring representation of a fixed, learned task.

85 **PPC representations facilitate a linear readout**

86 We asked whether precise task information can be extracted from this population of neurons, de-
87 spite the continual activity reconfiguration evident in these data. We began by fitting a linear de-
88 coder for each task variable of interest (animal location, heading, and velocity) for each day. This
89 model has the form $x(t)=M^T z(t)$, where $x(t)$ is the time-binned estimate of position, velocity or head-
90 ing (view angle) in the virtual maze, M is a vector of weights, and $z(t)$ is the normalized time-binned
91 calcium fluorescence (Methods: *Decoding analyses*).

92 Example decoding results for two mice are shown in Fig. 2a, and summaries of decoding per-

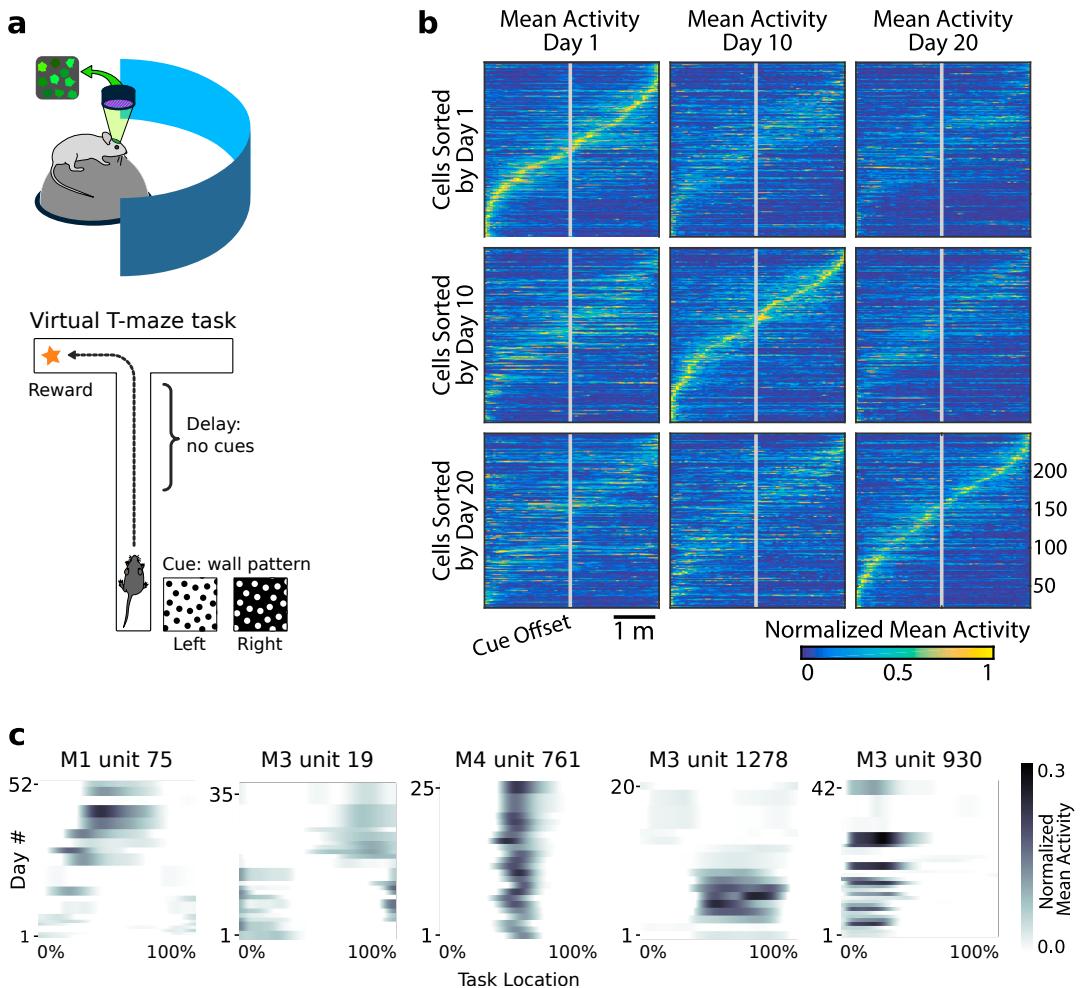


Figure 1. Neural population coding of spatial navigation reconfigures over time in a virtual-reality maze task (a) Mice were trained to use visual cues to navigate to a reward in a virtual-reality maze; neural population activity was recorded using Ca^{2+} imaging Driscoll et al. (2017). (b) (Reprinted from (Driscoll et al., 2017)) Neurons in PPC (vertical axes) fire at various regions in the maze (horizontal axes). Over days to weeks, individual neurons change their tuning, reconfiguring the population code. This occurs even at steady-state behavioral performance (after learning). (c) Each plot shows how location-averaged normalized activity changes for single cells over weeks. Missing days are interpolated to the nearest available sessions, and both left and right turns are combined. Neurons show diverse changes in tuning over days, including instability, relocation, long-term stability, gain/loss of selectivity, and intermittent responsiveness.

formance for four mice in Fig. 2b. Position, speed, and view angle can each be recovered with a separate linear model. The average mean absolute decoding error for all animals included in the analysis was $47.2 \text{ cm} \pm 8.8 \text{ cm}$ (mean ± 1 standard deviation) for position, $9.6 \text{ cm/s} \pm 2.2 \text{ cm/s}$ for speed, and $13.8^\circ \pm 4.0^\circ$ for view angle (Methods: *Decoding analyses*).

We chose a linear decoder specifically because it can be interpreted biologically as a single ‘read-out’ neuron that receives input from a few hundred PPC neurons, and whose activity approximates a linear weighted sum. The fact that a linear decoder recovers behavioral variables to reasonable accuracy suggests that brain areas with sufficiently dense connectivity to PPC can extract this information via simple weighted sums.

The number of PPC neurons recorded is a subset of the total PPC population. To assess whether additional neurons might improve decoding accuracy, we evaluated decoding performance of randomly drawn subsets of recorded neurons (Fig. 2c). Extrapolation of the decoding performance suggested that better performance might be possible with a larger population of randomly sam-

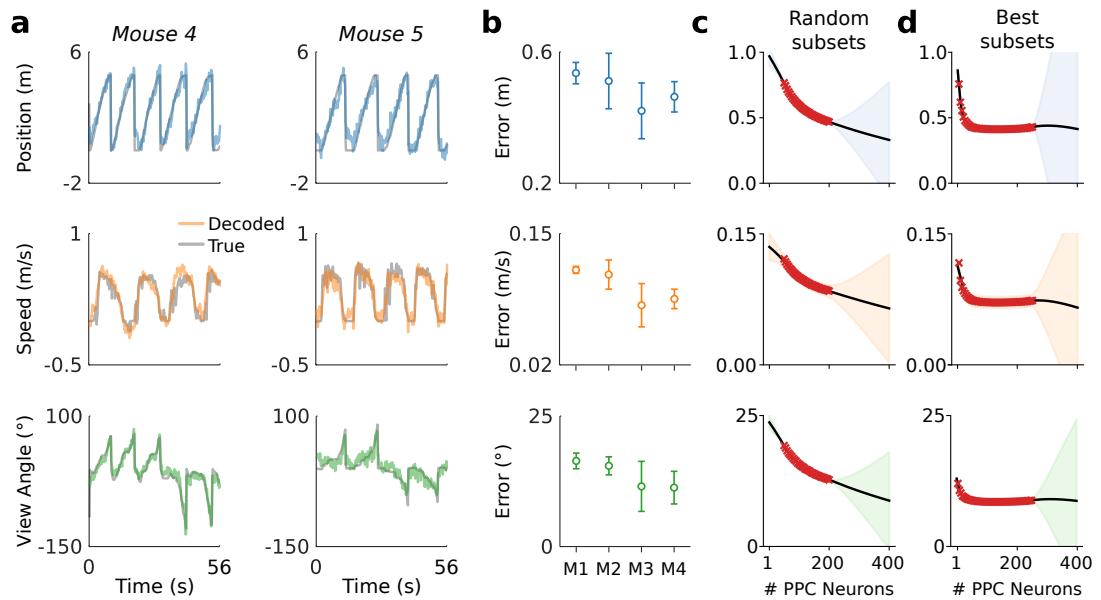


Figure 2. A linear decoder can extract kinematic information from PPC population activity on a single day. (a) Example decoding performance for a single session for mice 4 and 5. Grey denotes held-out test data; colors denote the prediction for the corresponding kinematic variable. (b) Summary of the decoding performance on single days; each point denotes one mouse. Error bars denote one standard deviation over all sessions that had at least $N=200$ high-confidence PPC neurons for each mouse. (Mouse 2 is excluded due to an insufficient number of isolated neurons). Chance level is ~ 1.5 m for forward position, and varies across subjects for forward velocity (~ 0.2 - 0.25 m/s) and head direction (~ 20 - 30 °). (c) Extrapolation of the performance of the static linear decoder for decoding position as a function of the number of PPC neurons, done via Gaussian process regression (Methods). Red "x" marks denote data; solid black line denotes the inferred mean of the GP. Shaded regions reflect $\pm 1.96\sigma$ Gaussian estimates of the 95th and 5th percentiles. (d) Same as panel (c), but where the neurons have been ranked such that the "best" subset of size $1 \leq K \leq N$ is chosen, selected by greedy search based on explained variance (Methods: Best K-Subset Ranking).

106 pleted PPC neurons than we recorded.

107 It is possible that a random sample of neurons misses the 'best' subset of cells for decoding task
108 variables. When we restricted to optimal subsets of neurons we found that performance improved
109 rapidly up to ~ 30 neurons and saturated at $\sim 30\%$ (50-100 neurons) of the neurons recorded (Fig. 2d).
110 On a given day task variables could be decoded well with relatively few (~ 10) neurons. However,
111 the identity of the neurons in this optimal subset changed over days. For all subjects, no more
112 than 1% of cells were consistently ranked in the top 10%, and no more than 13% in the top 50%.
113 We confirmed that this instability was not due to under-regularization in training (Methods: Best
114 K-Subset Ranking).

115 Of the neurons with strong location tuning, *Driscoll et al. (2017)* found that 60% changed their
116 location tuning over two weeks and a total of 80% changed over the 30 day period examined. We
117 find that even the small remaining 'stable' subset of neurons exhibited daily variations in their
118 Signal-to-Noise Ratio (SNR) with respect to task decoding, consistent with other studies (*Carmena
119 et al., 2005*). For example, no more than 8% of neurons that were in the top 25% in terms of
120 tuning-peak stability were also consistently in the top 25% in terms of SNR for all days. If a neuron
121 becomes relatively less reliable, then the weight assigned may become inappropriate for decoding.
122 This affects our analyses, and would also physiologically affect a downstream neuron with fixed
123 synaptic weights.

124 **Representational drift is systematic and significantly degrades a fixed readout**

125 Naively fitting a linear model to data from any given day shows that behavioural variables are en-
126 coded in a way that permits a simple readout, but there is no guarantee that this readout will sur-

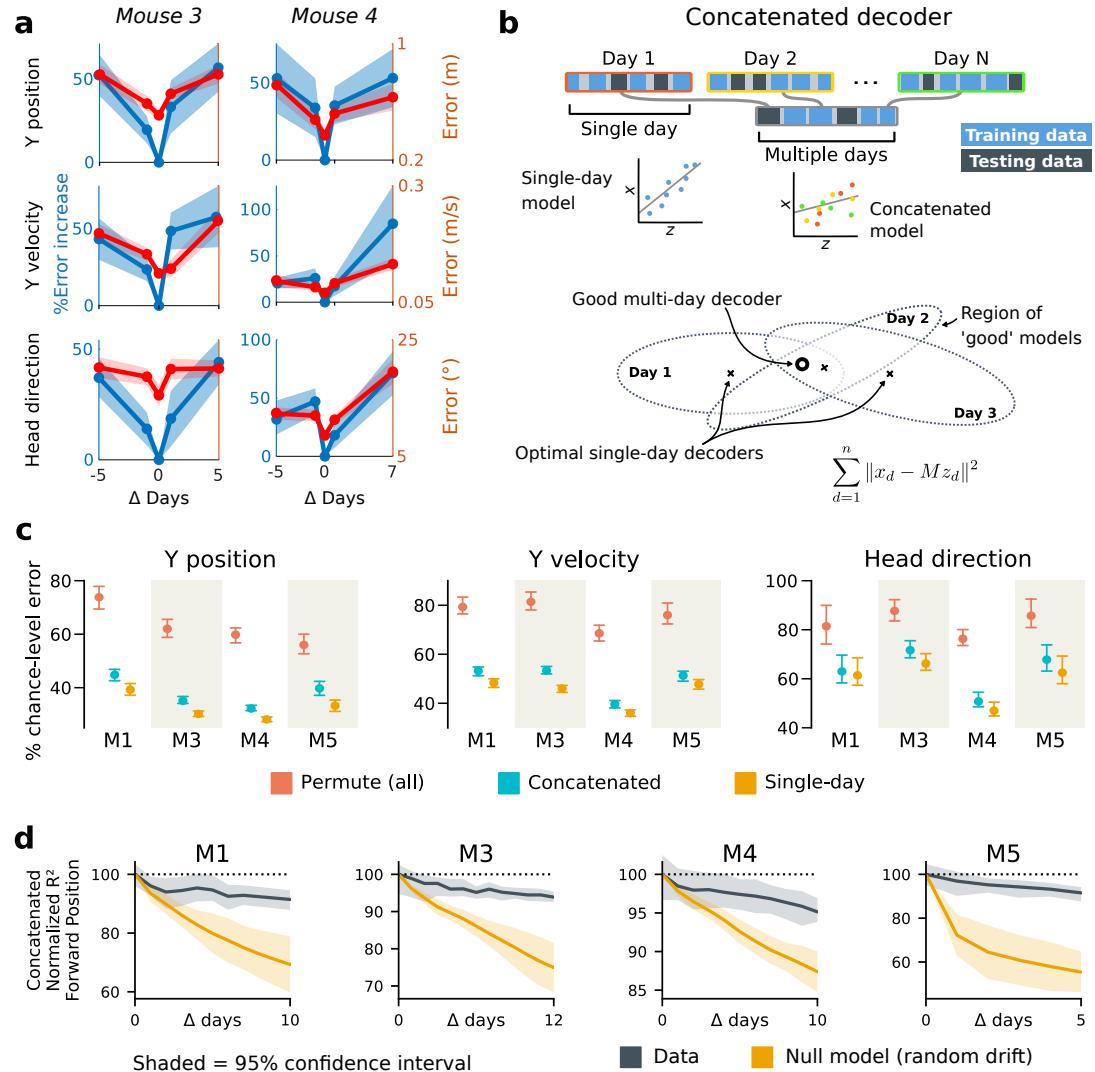


Figure 3. Single-day decoders generalize poorly to previous and subsequent days, but multi-day decoders exist with good performance. (a) Blue: % increase in error over the optimal decoder for the testing day (mouse 3, 136 neurons; mouse 4, 166 neurons). Red: Mean absolute error for decoders trained on a single day ('0') and tested on past/future days. (b) Fixed decoders M for multiple days $d \in 1 \dots D$ ('concatenated decoders') are fit to concatenated excerpts from several sessions. The inset equation reflects the objective function to be minimized (Methods). Due to redundancy in the neural code, many decoders can perform well on a single day. Although the single-day optimal decoders vary, a stable subspace with good performance can exist. (c) Concatenated decoders (cyan) perform slightly but significantly worse than single-day decoders (ochre; Mann-Whitney U test, $p < 0.01$). They also perform better than expected if neural codes were unrelated across days (permutation tests; red). Plots show the mean absolute decoding error as a percent of the chance-level error (points: median, whiskers: 5th-95th%). Chance-level error was estimated by shuffling kinematics traces relative to neural time-series (mean of 100 samples). For the permutation tests, 100 random samples were drawn with the neuronal identities randomly permuted. (d) Plots show the rate at which concatenated-decoder accuracy (normalized R^2) degrades as the number of days increase. Concatenated decoders (black) degrade more slowly than expected for random drift (ochre). Shaded regions reflect the inner 95% of the data (generated by resampling for the null model). The null model statistics are matched to the within- and between-day variance and sparsity of the experimental data for each animal (Methods).

127 vive long-term drift in the neural code. To illustrate this we compared the decoding performance
 128 of models fitted on a given day with decoders optimized on data from earlier or later days. We
 129 restricted this analysis to those neurons that were identified with high confidence on all days con-
 130 sidered. We found that decoding performance decreased as the separation between days grew
 131 (Fig 3a). This is unsurprising given the extent of reconfiguration reported in the original study
 132 (*Driscoll et al., 2017*) and depicted in Fig 1. Furthermore, because task-related PPC activity is dis-
 133 tributed over many neurons, many different linear decoders can achieve similar error rates due
 134 to the degeneracy in the representation (*Rokni et al., 2007; Kaufman et al., 2014; Montijn et al.,*
 135 *2016*). Since the directions in population activity used for inter-area communication might differ
 136 from the directions that maximally encode stimulus information in the local population (*Ni et al.,*
 137 *2018; Semedo et al., 2019*), single-day decoders might overlook a long-term stable subspace used
 138 for encoding and communication. This motivates the question of whether a drift-invariant linear
 139 decoder exists and whether its existence is biologically plausible.

140 To address this, we tested the performance of a single linear decoder optimized across data
 141 from multiple days. We concatenated data from different days using the same subset of PPC neu-
 142 rons (Fig. 3b). In all four subjects, we found that such fixed multiple-day linear ‘concatenated’ de-
 143 coders could recover accurate task variable information despite ongoing changes in PPC neuron
 144 tuning. However, the average performance of the multiple-day decoders was significantly worse
 145 than single-day linear decoders for each day (Fig. 3c).

146 The existence of a fixed, approximate decoder implies a degenerate representation of task vari-
 147 ables in the population activity of PPC neurons. In other words, there is a family of linear decoders
 148 that can recover behavioral variables while allowing weights to vary in some region of weight space.
 149 This situation is illustrated in Figure 3b, which depicts regions of good performance of single-day
 150 linear decoders as ellipsoids. The existence of an approximate concatenated decoder implies that
 151 these ellipsoids intersect over several days for some allowable level of error in the decoder. For a
 152 sufficiently redundant neural code, one might expect to find an invariant decoder for some speci-
 153 fied level of accuracy even if the underlying code drifts. However, there are many qualitative ways
 154 in which drift can occur in a neural code: it could resemble a random walk, as some studies suggest
 155 (*Moczulska et al., 2013; Loewenstein et al., 2011, 2015*), or there could be a systematic component.
 156 Is the accuracy we observe in the concatenated decoder expected for a random walk? In all sub-
 157 jects, we found that a concatenated decoder performed substantially better on experimental data
 158 than on randomly drifting synthetic data with matched sparseness and matched within/between-
 159 session variability (Fig. 3d). This suggests that the drift in the neural data is not purely random.

160 We further investigated the dynamics of drift by quantifying the direction of changes in neural
 161 variability over time (Fig. 4c,d, Methods: Drift alignment). We found that drift is indeed aligned
 162 above chance to within-session neural population variability. This suggests that the biological
 163 mechanisms underlying drift are in part systematic and constrained by a requirement to keep a
 164 consistent population code over time. In comparison, the projection of drift onto behavior-coding
 165 directions was small, but still above chance. This is consistent with the hypothesis that ongoing
 166 compensation might be needed for a long-term stable readout.

167 To quantify the systematic nature of drift further, we modified the null model to make drift
 168 partially systematic by constraining the null-model drift within a low rank subspace (Fig. 4-S1). This
 169 reflects a scenario in which only a few components of the population code change over time. We
 170 found that the performance of a concatenated decoder for low-rank drift better approximated
 171 experimental data. For three of the four mice we could match concatenated decoder performance
 172 when the dimension of the drift process was constrained within a range of 14-26, a relatively small
 173 fraction (around 20%) of the components of the full population.

**174 Biologically achievable rates of plasticity can compensate drift, independent of
175 specific learning rules**

176 Together, these analyses show that the observed dynamics of drift favor a fixed linear readout
177 above what would be expected for random drift. However, our results also show that a sub-
178 stantial component of drift cannot be confined to the null space of a fixed downstream linear
179 readout. We asked how much ongoing weight change would be needed to achieve the perfor-
180 mance of single-day decoders while minimizing day-to-day changes in decoding weights. We first
181 approached this without assuming a specific plasticity rule, by simultaneously optimizing linear
182 decoders for all recorded days while penalizing the magnitude of weight change between ses-
183 sions (Fig. 4a, Methods: *Concatenated and constrained analyses*). By varying the magnitude of the
184 weight change penalty we interpolated between the concatenated decoder (no weight changes)
185 and the single-day decoders (optimal weights for each day). The result of this is shown in Figure
186 4b. Performance improves rapidly once small weight changes are permitted (~12-25% per session).
187 Thus, relatively modest amounts of synaptic plasticity might suffice to keep encoding consistent
188 with changes in representation, provided a mechanism exists to implement appropriate weight
189 changes.

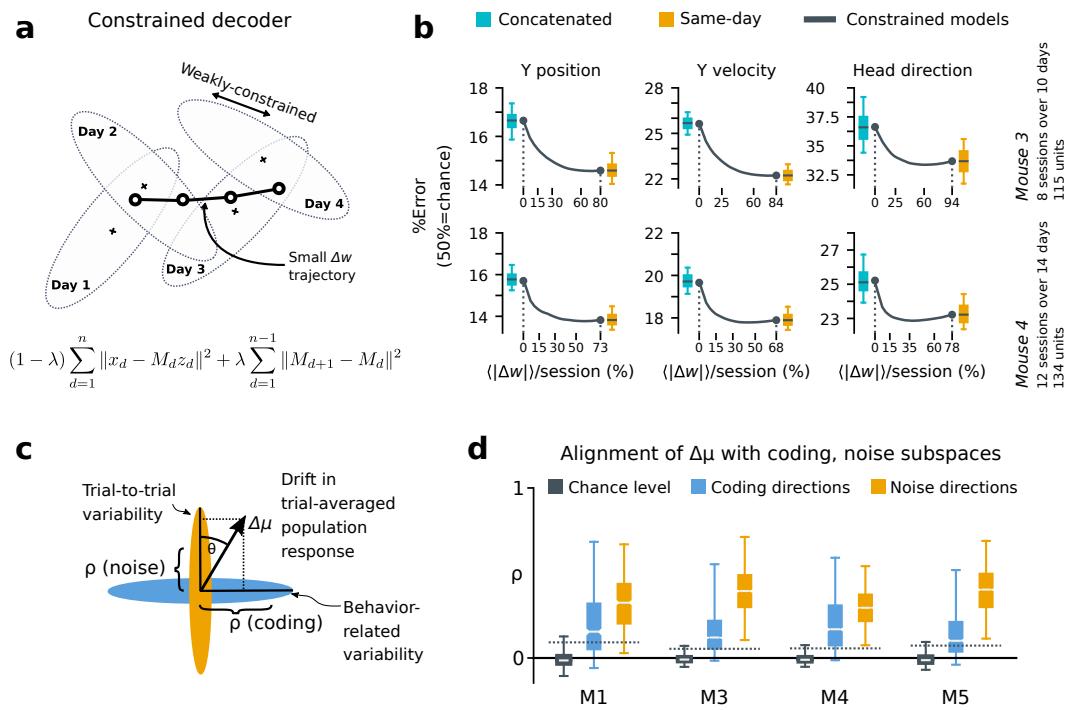


Figure 4. A slowly-varying component of drift disrupts the behavior-coding subspace. (a) The small error increase when training concatenated decoders (Fig. 3) suggests that plasticity is needed to maintain good decoding in the long term. We assess the minimum rate for this plasticity by training a separate decoder M_d for each day, while minimizing the change in weights across days. The parameter λ controls how strongly we constrain weight changes across days (the inset equation reflects the objective function to be minimized; Methods). (b) Decoders trained on all days (cyan) perform better than chance (red), but worse than single-day decoders (ochre). Black traces illustrate the plasticity-accuracy trade-off for adaptive decoding. Modest weight changes per day are sufficient to match the performance of single-day decoders (Boxes: inner 50% of data, horizontal lines: median, whiskers: 5–95th%). (c) Across days, the mean neural activity associated with a particular phase of the task changes ($\Delta\mu$). We define an alignment measure ρ (Methods) to assess the extent to which these changes align with behavior-coding directions in the population code (blue) versus directions of noise correlations (orange). (d) Drift is more aligned with noise (orange) than it is with behavior-coding directions (blue). Nevertheless, drift overlaps this behavior-coding subspace much more than chance (grey; dashed line: 95% Monte-Carlo sample). Each box reflects the distribution over all maze locations, with all consecutive pairs of sessions combined.

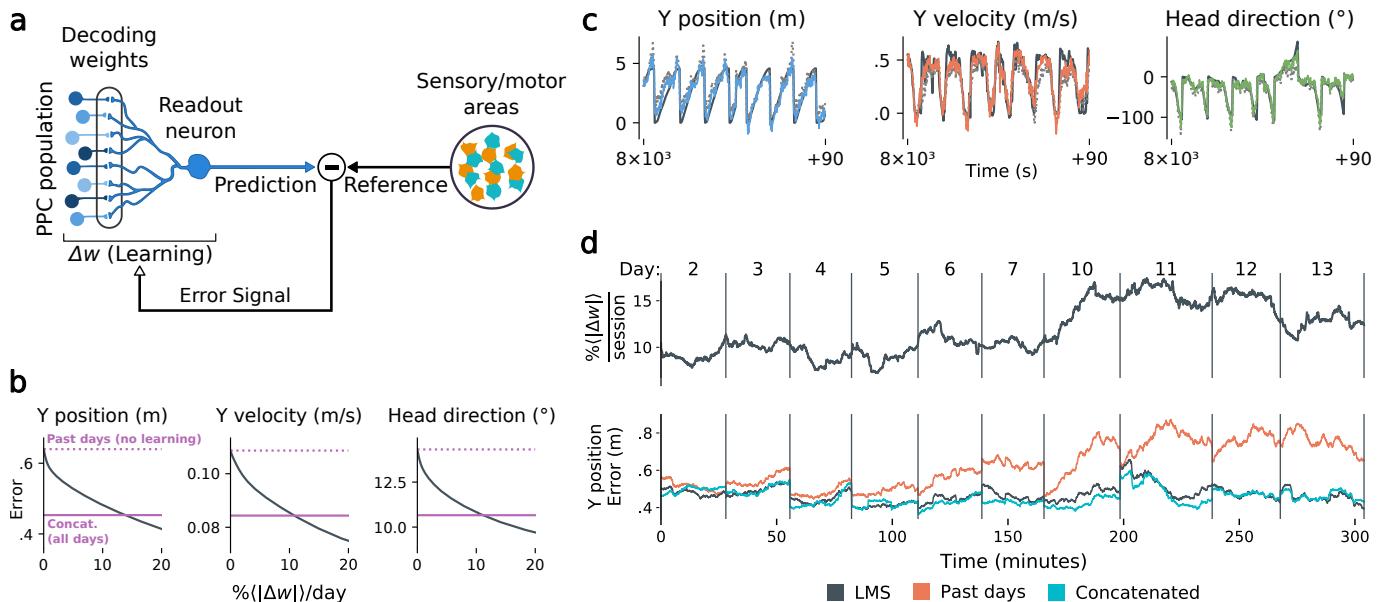


Figure 5. Local, adaptive decoders can track representational drift over multiple days. (a) The Least Mean-Squares (LMS) algorithm learns to linearly decode a target kinematic variable based on error feedback. Continued online learning can track gradual reconfiguration in population representations. (b) As the average weight change per day (horizontal axis) increases, the average decoding error (vertical axis) of the LMS algorithm improves, shown here for three kinematic variables (Mouse 4, 144 units, 10 sessions over 12 days; Methods: Online LMS algorithm). (Dashed line: error for a decoder trained on only the previous session without online learning; Solid line: performance of a decoder trained over all testing days). As the rate of synaptic plasticity is increased, LMS achieves error rates comparable to the concatenated decoder. (c) Example LMS decoding results for three kinematic variables. Ground truth is plotted in black, and LMS estimate in color. Sample traces are taken from day six. Dashed traces indicate the performance of the decoder without ongoing re-training. (d) (top) Average percent weight-change per session for online decoding of forward position (learning rate: 4×10^{-4} /sample). The horizontal axis reflects time, with vertical bars separating days. The average weight change is 10.2% per session. To visualize $\%|\Delta w|$ continuously in this plot, we use a sliding difference with a window reflecting the average number of samples per session. (bottom) LMS (black) performs comparably to the concatenated decoder (cyan) (LMS mean absolute error of 0.47 m is within $\leq 3\%$ of concatenated decoder error). Without ongoing learning, the performance of the initial decoder degrades (orange). Error traces have been averaged over ten minute intervals within each session. Discontinuities between days reflect day-to-day variability and suggest a small transient increase in error for LMS decoding at the start of each day.

190 A biologically plausible local learning rule can compensate drift

191 The results in Figure 4b suggest that modest amounts of synaptic plasticity could compensate for
 192 drift, but do not suggest a biologically plausible mechanism for this compensation. Could neu-
 193 rons track slow reconfiguration using locally available signals in practice? To test this, we used an
 194 adaptive linear neuron model based on the least mean square learning (LMS) rule (*Widrow and*
 195 *Hoff, 1960, 1962*) (Methods). This algorithm is biologically plausible because it only requires each
 196 synapse to access its current weight and recent prediction error (Fig. 5a, Methods: *Online LMS*
 197 *algorithm*).

198 Fig. 5b shows that this online learning rule achieved decoding performance comparable to
 199 the offline constrained decoders. Over the timespan of the data, LMS allows a linear decoder to
 200 track representational drift observed (Fig. 5c), exhibiting weight changes of $\sim 10\%/\text{day}$ across all
 201 animals (learning rate $4 \times 10^{-4}/\text{sample}$, Fig. 5-S1). These results suggest that small weight changes
 202 could track representational drift in practice. In contrast, we found that LMS struggled to match
 203 the unconstrained drift of the null model explored in Figure 3d. Calibrating the LMS learning rate
 204 on the null model to match the mean performance seen on the true data required an average

205 weight change of 93% per day. In comparison, matching the average percent weight change per
 206 day of 10%, the null model produced a normalized mean-squared-error of $1.3\sigma^2$ (averaged over all
 207 mice), worse than chance. This further indicates that drift is highly structured, facilitating online
 208 compensation with a local learning rule.

209 We stress that modelling assumptions mean that these results are necessarily a proxy for the
 210 rates of synaptic plasticity that are observed *in vivo*. Nonetheless, we believe these calculations are
 211 conservative. We were restricted to a sample of ~100-200 neurons, at least an order of magnitude
 212 less than the typical number of inputs to a pyramidal cell in cortex. The per-synapse magnitude
 213 of plasticity necessarily increases when smaller subsets are used for a readout (Fig. 5-S2). One
 214 would therefore expect lower rates of plasticity for larger populations. Indeed, when we combined
 215 neurons across mice into a large synthetic population (1238 cells), we found that the plasticity
 216 required to achieve target error asymptotes at less than 4% per day (Fig. 5-S3). Together, these
 217 results show a conservatively achievable bound on the rate of plasticity required to compensate
 218 drift in a biologically plausible model.

219 Discussion

220 Several theories have been proposed for how stable behavior could be maintained despite on-
 221 going changes in connectivity and neural activity. Here, we found that representational drift oc-
 222 curred in both coding and non-coding subspaces. On a timescale of a few days, redundancy in
 223 the neural population could accommodate a significant component of drift, assuming a biological
 224 mechanism exists for establishing appropriate readout weights. Simulations suggested that the ex-
 225 istence of this approximately stable subspace were not simply a result of population redundancy,
 226 since random diffusive drift quickly degraded a downstream readout. Drift being confined to a low-
 227 dimensional subspace is one scenario that could give rise to this, although we do not exclude other
 228 possibilities. Nevertheless, a non-negligible component of drift resides outside the null space of a
 229 linear encoding subspace, implying that drift will eventually destroy any fixed-weight readout.

230 However, we showed that this destructive component of drift could be compensated with small
 231 and biologically realistic changes in synaptic weights, independently of any specific learning rule.
 232 Furthermore, we provided an example of a simple and biologically plausible learning rule that can
 233 achieve such compensation over long timescales with modest rates of plasticity. If our modeling
 234 results are taken literally, this would suggest that a single unit with connections to ~100 PPC neu-
 235 rons can accurately decode task information with modest changes in synaptic weights over many
 236 days. This provides a concrete and quantitative analysis of the implications of drift on synaptic
 237 plasticity and connectivity. Together, our findings provide some of the first evidence from experi-
 238 mental data that representational drift could be compatible with long-term memories of learned
 239 behavioral associations.

240 A natural question is whether a long-term stable subspace is supported by an unobserved sub-
 241 set of neurons that have stable tuning (*Clopath et al., 2017*). We do not exclude this possibility
 242 because we measured a subset of the neural population. However, over multiple samples from
 243 different animals our analyses consistently suggest that drift will reconfigure the code entirely over
 244 months. Specifically, we found that past reliability in single cells is no guarantee of future stability.
 245 This, combined with an abundance of highly-informative cells on a single day, contributes to poor
 246 (fixed) decoder generalization, because previously reliable cells eventually drop out or change their
 247 tuning. Consistent with this, studies have shown that connectivity in mammalian cortex is surpris-
 248 ingly dynamic. Connections between neurons change on a timescale of hours to days with a small
 249 number of stable connections (*Holtmaat et al., 2005; Minerbi et al., 2009; Holtmaat and Svoboda,*
2009; Attardo et al., 2015).

250 We stress that the kind of reconfiguration observed in PPC is not seen in all parts of the brain;
 251 primary sensory and motor cortices can show remarkable stability in neural representations over
 252 time (*Gallego et al., 2020*). However, even if stable representations exist elsewhere in the brain,
 254 PPC still must communicate with these areas. We suggest that a component of ongoing plasticity

255 maintains congruent representations across different neural circuits. Such maintenance would be
 256 important in a distributed, adaptive system like the brain, in which multiple areas learn in parallel.
 257 How this is achieved is the subject of intense debate (*Rule et al., 2019*). We hypothesize that
 258 neural circuits have continual access to two kinds of error signals. One kind should reflect mis-
 259 match between internal representations and external task variables, and another should reflect
 260 prediction mismatch between one neural circuit and another. Our study therefore motivates new
 261 experiments to search for neural correlates of error feedback between areas, and suggests further
 262 theoretical work to explore the consequences of such feedback.

263 Acknowledgements

264 We thank Fulvio Forni, Yaniv Ziv and Alon Rubin for in depth discussions. This work was supported
 265 by the Human Frontier Science Program (RGY0069), ERC Starting Grant (StG FLEXNEURO 716643)
 266 and grants from the NIH (NS089521, MH107620, NS108410)

267 Declaration of Interests

268 The authors declare no conflict of interest.

269 Methods

270 Data acquisition

271 The behavioral and two-photon calcium imaging data analyzed here was provided by the Harvey
 272 lab. Details regarding the experimental subjects and methods are provided in *Driscoll et al. (2017)*.

273 Virtual reality task

274 Details of the virtual reality environment, training protocol, and fixed association navigation task
 275 are described in *Driscoll et al. (2017)*. In brief, virtual reality environments were constructed and
 276 operated using the MATLAB-based ViRMEn software (Virtual Reality Mouse Engine) *Harvey et al.*
 277 (2012). Data were obtained from mice that had completed the 4-8 week training program for the
 278 two-alternative forced choice T-maze task. The length of the virtual reality maze was fixed to have
 279 a total length of 4.5 m. The cues were patterns on the walls (black with white dots or white with
 280 black dots), and were followed by a gray striped 'cue recall' segment (2.25 m long) that was identical
 281 across trial types.

282 Data preparation and pre-processing

283 Raw Ca^{2+} fluorescence videos (sample rate=5.3 Hz) were corrected for motion artefacts, and in-
 284 dividual sources of Ca^{2+} fluorescence were identified and extracted (*Driscoll et al., 2017*). Pro-
 285 cessed data consisted of normalized Ca^{2+} fluorescence transients (" $\Delta F/F$ ") and behavioral vari-
 286 ables (mouse position, view angle, and velocity). Inter-trial intervals (ITIs) were removed for all
 287 subsequent analyses. For offline decoding, we considered only correct trials, and all signals were
 288 centered to zero-mean on each trial as a pre-processing step.

289 When considering sequences of days, we restricted analysis to units that were continuously
 290 tracked over all days. For figures 3 and 4, we use the following data: M1: 7 sessions, 15 days, 101
 291 neurons; M3: 10 sessions, 13 days, 114 neurons; M4: 10 sessions, 11 days, 146 neurons; M5: 7
 292 sessions, 7 days, 112 neurons. We allowed up to two-day recording gaps between consecutive
 293 sessions from the same mouse.

294 QUANTIFICATION AND STATISTICAL ANALYSIS

295 Decoding analyses

296 We decoded kinematics time-series $\mathbf{x}=\{x_1, \dots, x_T\}$ with T time-points from the vector of instantan-
 297 eous neural population activity $\mathbf{z}=\{z_1, \dots, z_T\}$, using a linear decoder with a fixed set of weights
 298 M , i.e. $\hat{\mathbf{x}} = M^T \mathbf{z}$. We used the ordinary least-squares (OLS) solution for M , which minimizes the
 299 squared (L2) prediction error $\varepsilon = \|\mathbf{x} - M^T \mathbf{z}\|^2$ over all time-points. For the 'same-day' analyses, we

300 optimize a separate M_d for each day d (Fig. 2), restricting analysis to sessions with at least 200
 301 identified units. We assessed decoding performance using 10-fold cross-validation, and report the
 302 mean absolute error, defined as $\langle |\mathbf{x} - \hat{\mathbf{x}}| \rangle$. Here, $|\cdot|$ denotes the element-wise absolute value, and
 303 $\langle \cdot \rangle$ denotes expectation.

304 Best K-Subset Ranking

305 For Fig. 2d, we ranked cells in order of explained variance using a greedy algorithm. Starting with
 306 the most predictive cell, we iteratively added the next cell that minimized the MSE under ten-fold
 307 cross-validated linear decoding. To accelerate this procedure, we pre-computed the mean and
 308 covariance structure for training and testing datasets. MSE fits and decoding performance can be
 309 computed directly from these summary statistics, accelerating the several thousand evaluations
 310 required for greedy selection. We added L2 regularization to this analysis by adding a constant λI
 311 to the covariance matrix of the neural data. The optimal regularization strength ($\lambda = 10^{-4}$ to 10^{-3})
 312 slightly reduced decoding error, but did not alter the ranking of cells.

313 Extrapolation via GP regression

314 To qualitatively assess whether decoding performance saturates with the available number of
 315 recorded neurons, we computed decoding performance on a sequence of random subsets of the
 316 population of various sizes (Fig. 2c,d). Results for all analyses are reported as the mean over 20
 317 randomly-drawn neuronal sub-populations, and over all sessions that had at least $N=150$ units.
 318 Gaussian process (GP) regression was implemented in Python, using a combination of a Matérn
 319 kernel and an additive white noise kernel. Kernel parameters were optimized via maximum likeli-
 320 hood (Scikit-learn, *Pedregosa et al. (2011)*).

321 Concatenated and constrained analyses

322 For both the concatenated (Fig. 3b,e) and constrained analyses (Fig. 4a,b), we used the set of
 323 identified neurons included in all sessions considered. In the concatenated analyses, we solved
 324 for a single decoder M_c for all days:

$$\varepsilon = \sum_{d=1}^n \|\mathbf{x}_d - M_c^\top \mathbf{z}_d\|^2, \quad (1)$$

325 where ε denotes the quadratic objective function to be minimized. In the constrained analysis,
 326 we optimized a series of different weights $\mathbf{M}=\{M_1, \dots, M_D\}$ for each day $d \in 1..D$, and added an ad-
 327 justable L2 penalty λ on the change in weights across days. This problem reduces to the 'same-day'
 328 analysis for $\lambda=0$, and approaches the concatenated decoder as λ approaches 1:

$$\varepsilon = (1 - \lambda) \sum_{d=1}^n \|\mathbf{x}_d - M_d^\top \mathbf{z}_d\|^2 + \lambda \sum_{d=1}^{n-1} \|M_{d+1} - M_d\|^2. \quad (2)$$

329 For the purposes of the constrained analysis, missing days were ignored and the remaining days
 330 treated as if they were contiguous. Two sessions were missing from the 10 and 14-day spans for
 331 mice 3 and 4, respectively (Fig. 4b). Figure 3c also shows the expected performance of a con-
 332 catenated decoder for completely unrelated neural codes. To assess this, we permuted neuronal
 333 identities within individual sessions, so that each day uses a different "code".

334 **Null model**

335 We developed a null model to assess whether the performance of the concatenated decoder was
 336 consistent with random drift. For this, we matched the amount of day-to-day drift based on the
 337 rate at which single-day decoders degrade. We also sampled neural states from the true data in
 338 order to preserve sparsity and correlation statistics. The null model related neural activity to a
 339 'fake' observable readout (e.g. mouse position) via an arbitrary linear mapping. The null model
 340 changed from day to day, reflecting drift in the neural code. The fidelity of single day and across
 341 day decoders in inferring a readout from the null model was matched to the true data.

342 For each animal we take the matrix $z \in \mathbb{R}^{n \times d}$ of mean-centered neural activity on day one, where
 343 n represents the number of recorded neurons and d represents the number of datapoints. We relate
 344 this matrix to pseudo-observations of mouse position z via a null model of the form $z_r = M_r^T z + \epsilon_r$,
 345 where $M_r^T, \epsilon_r \in \mathbb{R}^{1 \times n}$. Note that r indexes days. The vector ϵ_r is generated as scaled i.i.d. Gaussian
 346 noise. We scale ϵ_r such that the accuracy of a linear decoder trained on the data (z, x_r) matches the
 347 average (over days) accuracy of a single-day decoder trained on the true data.

348 Next, we consider the choice of the randomly-drifting readout, M_r . On day one, M_1 is generated
 349 as a vector of uniform random variables on $[0, 1]$. Given M_r , we desire an M_{r+1} that satisfies

- 350 • $\|M_{r+1}\|_2 = \|M_r\|_2$.
- 351 • The expected coefficient of multiple correlation of $x_{r+1} = M_{r+1}^T z$ against the predictive model
 352 $M_r^T z$ (between day R^2) matches the average (over days) of the equivalent statistic generated
 353 from the true data.

354 To do this, we first generate a candidate $\Delta M' \in \mathbb{R}^{n \times 1}$ as a vector of i.i.d. white noise. The components
 355 of $\Delta M'$ orthogonal and parallel to M_r are then scaled so that $M_{r+1} = M_r + \Delta M'$ satisfies the
 356 constraints above.

357 In Figure 4-S1, a modification of the null model that confined inter-day model drift to a pre-
 358 defined subspace was used. Before simulating the null model over days, we randomly chose k
 359 orthogonal basis vectors, representing a k -dimensional subspace. We then searched for a candi-
 360 date $\Delta M'_r$, on each inter-day interval, that was representable as a weighted sum of these basis
 361 vectors. This requirement was in addition to those previously posed. Finding such a $\Delta M'_r$ corre-
 362 sponds to solving a quadratically-constrained quadratic program. This is non-convex, and thus a
 363 solution will not necessarily be found. However, solutions were always found in practice. We used
 364 unit Gaussian random variables as our initial guesses for each component of $\Delta M'_r$, before solving
 365 the quadratic program using the IPOPT toolbox (*Wächter and Biegler, 2006*).

366 Drift alignment

367 We examine how much drift aligns with noise correlations versus directions of neural activity that
 368 vary with the task ("behavior-coding directions"). We define an alignment statistic ρ that reflects
 369 how much drift projects onto a given subspace (i.e. noise vs. behavior). We normalize ρ so that 0
 370 reflects chance-level alignment and 1 reflects perfect alignment of the drift with the largest eigen-
 371 vector of a given subspace (e.g. the principal eigenvector of the noise covariance).

372 Let $z(x)$ denote the neural population activity, where x reflects a normalized measure of maze
 373 location, akin to trial pseudotime. Define drift $\Delta \mu_z(x)$ as the change in the mean neural activity $\mu_z(x)$
 374 across days. We examine how much drift aligns with noise correlations versus directions of neural
 375 activity that vary with task pseudotime ($dz(x)/dx$).

376 To measure the alignment of a drift vector $\Delta \mu$ with the distribution of inter-trial variability (i.e.
 377 noise), we consider the trial-averaged mean μ and covariance Σ of the neural activity (log calcium-
 378 fluorescence signals filtered between 0.03 and .3 Hz and z-scored), conditioned on trial location
 379 and the current/previous cue direction. We use the mean squared magnitude of the dot product
 380 between the change in trial-conditioned means between days ($\Delta \mu$), with the directions of inter-trial
 381 variability ($\Delta z = z - \langle z \rangle$) on the first day, which is summarized by the product $\Delta \mu^T \Sigma \Delta \mu$:

$$\begin{aligned} \langle |\Delta \mu^T \Delta z|^2 \rangle &= \langle \Delta \mu^T \Delta z \Delta z^T \Delta \mu \rangle \\ &= \Delta \mu^T \langle \Delta z \Delta z^T \rangle \Delta \mu \\ &= \Delta \mu^T \Sigma \Delta \mu. \end{aligned} \tag{3}$$

382 To compare pairs of sessions with different amounts of drift and variability, we normalize the
 383 drift vector to unit length, and normalize the trial-conditioned covariance by its largest eigenvalue
 384 λ_{\max} :

$$\phi_{\text{trial}}^2 = \frac{\Delta \mu^T \Sigma \Delta \mu}{|\Delta \mu|^2 \cdot \lambda_{\max}} \tag{4}$$

385 The statistic ϕ_{trial} equals 1 if the drift aligns perfectly with the direction of largest inter-trial variability,
 386 and can be interpreted as the fraction of drift explained by the directions of noise correlations.
 387 Random drift can still align with some directions by chance, and the mean squared dot-product
 388 between two randomly-oriented D -dimensional unit vectors scales as $1/D$. Accounting for the
 389 contribution from each dimension of Σ , the expected chance alignment is therefore $\phi_0^2 = \text{tr}(\Sigma)/(D \cdot$
 390 $\lambda_{\max})$. We normalize the alignment coefficient ρ_{noise} such that it is 0 for randomly oriented vectors,
 391 and 1 if the drift aligns perfectly with the direction of largest variability:

$$\rho_{\text{noise}} = \frac{\phi_{\text{trial}} - \phi_0}{1 - \phi_0} \quad (5)$$

392 We define a similar alignment statistic ρ_{coding} to assess how drift aligns with directions of neural
 393 variability that encode location. We consider the root-mean-squared dot product between the drift
 394 $\Delta\mu$, and the directions of neural activity (z) that vary with location (x) on a given trial, i.e. $\nabla_x z(x)$:

$$\begin{aligned} \langle |\Delta\mu^\top \nabla_x z(x)|^2 \rangle &= \langle \Delta\mu^\top [\nabla_x z(x)][\nabla_x z(x)]^\top \Delta\mu \rangle \\ &= \Delta\mu^\top \langle [\nabla_x z(x)][\nabla_x z(x)]^\top \rangle \Delta\mu \\ &= \Delta\mu^\top [\Sigma_\nabla + \mu_\nabla \mu_\nabla^\top] \Delta\mu \end{aligned} \quad (6)$$

395 In contrast to the trial-to-trial variability statistic, this statistic depends on the second moment $\Sigma_\nabla +$
 396 $\mu_\nabla \mu_\nabla^\top$, where $\nabla_x z(x) \sim \mathcal{N}(\mu_\nabla, \Sigma_\nabla)$. We define a normalized ϕ_{coding}^2 and ρ_{coding} similarly to ϕ_{trial}^2 and ρ_{noise} .
 397 For the alignment of drift with behavior, we observed $\rho_{\text{coding}} = 0.11\text{--}0.24$ ($\mu=0.15$, $\sigma=0.03$), which
 398 was significantly above chance for all mice. In contrast, the 95th percentile for chance alignment
 399 (i.e. random drift) ranged from 0.06–0.10 ($\mu=0.07$, $\sigma=0.02$). Drift aligned substantially more with
 400 noise correlations, with $\rho=0.29\text{--}0.43$ ($\mu=0.36$, $\sigma=0.04$).

401 **Online LMS algorithm**

402 The Least Mean-Squares (LMS) algorithm is an online approach to training and updating a linear
 403 decoder, and corresponds to stochastic gradient-descent (Fig. 4a). The algorithm was originally
 404 introduced in (Widrow and Hoff, 1960, 1962; Widrow and Stearns, 1985). Briefly, LMS computes a
 405 prediction error for an affine decoder (i.e. a linear decoder with a constant offset feature or bias
 406 parameter) at every time-point, which is then used to update the decoding weights. We analyzed
 407 twelve contiguous sessions from mouse 4 (144 units in common), and initialized the decoder by
 408 training on the first two sessions using OLS.

409 By varying the learning rate, we obtained a trade-off (Fig. 4b) between the rate of weight
 410 changes and the decoding error, with the most rapid learning rates exceeding the performance
 411 of offline (static) decoders. In Fig. 4d, we selected an example with a learning rate of $\eta=4\times 10^{-4}$. To
 412 provide a continuous visualization of the rate of weight change in Fig. 4d, we used a sliding differ-
 413 ence with a duration matching the average session length. This was normalized by the average
 414 weight magnitude to report percent weight change per day. In all other statistics, per-day weight
 415 change is assessed as the difference in weights at the end of each session, divided by the days
 416 between the sessions.

417 **DATA AND CODE AVAILABILITY**

418 Datasets recorded in Driscoll et al. (Driscoll et al., 2017) are available from the Dryad repository
 419 (doi.org/10.5061/dryad.gqnk98sjq). The analysis code generated during this study is available on
 420 Github (github.com/michaelerule/stable-task-information).

421 **References**

422 Ajemian R, D'Ausilio A, Moorman H, Bizzi E. A theory for how sensorimotor skills are learned and re-
 423 tained in noisy and nonstationary neural circuits. Proceedings of the National Academy of Sciences. 2013;
 424 110(52):E5078–E5087.

- 425 **Andersen RA**, Buneo CA. Intentional maps in posterior parietal cortex. Annual review of neuroscience. 2002; 25(1):189–220.
- 427 **Andersen RA**, Snyder LH, Bradley DC, Xing J. Multimodal representation of space in the posterior parietal cortex and its use in planning movements. Annual review of neuroscience. 1997; 20(1):303–330.
- 429 **Attardo A**, Fitzgerald JE, Schnitzer MJ. Impermanence of dendritic spines in live adult CA1 hippocampus. Nature. 2015; 523(7562):592.
- 431 **Calton JL**, Taube JS. Where am I and how will I get there from here? A role for posterior parietal cortex in the integration of spatial information and route planning. Neurobiology of learning and memory. 2009; 91(2):186–196.
- 434 **Carmena JM**, Lebedev MA, Henriquez CS, Nicolelis MA. Stable ensemble performance with single-neuron variability during reaching movements in primates. Journal of Neuroscience. 2005; 25(46):10712–10716.
- 436 **Clopath C**, Bonhoeffer T, Hübener M, Rose T. Variance and invariance of neuronal long-term representations. Philosophical Transactions of the Royal Society B: Biological Sciences. 2017; 372(1715):20160161.
- 438 **Degenhart AD**, Bishop WE, Oby ER, Tyler-Kabara EC, Chase SM, Batista AP, Byron MY. Stabilization of a brain-computer interface via the alignment of low-dimensional spaces of neural activity. Nature Biomedical Engineering. 2020; p. 1–14.
- 441 **Driscoll LN**, Pettit NL, Minderer M, Chettih SN, Harvey CD. Dynamic reorganization of neuronal activity patterns in parietal cortex. Cell. 2017; 170(5):986–999.
- 443 **Druckmann S**, Chklovskii DB. Neuronal circuits underlying persistent representations despite time varying activity. Current Biology. 2012; 22(22):2095–2103.
- 445 **Gallego JA**, Perich MG, Chowdhury RH, Solla SA, Miller LE. Long-term stability of cortical population dynamics underlying consistent behavior. Nature Neuroscience. 2020; p. 1–11.
- 447 **Gallego JA**, Perich MG, Naufel SN, Ethier C, Solla SA, Miller LE. Cortical population activity within a preserved neural manifold underlies multiple motor behaviors. Nature communications. 2018; 9(1):4233.
- 449 **Ganguly K**, Carmena JM. Emergence of a stable cortical map for neuroprosthetic control. PLoS biology. 2009; 7(7):e1000153.
- 451 **Harvey CD**, Coen P, Tank DW. Choice-specific sequences in parietal cortex during a virtual-navigation decision task. Nature. 2012; 484(7392):62–68.
- 453 **Hennig JA**, Golub MD, Lund PJ, Sadtler PT, Oby ER, Quick KM, Ryu SI, Tyler-Kabara EC, Batista AP, Byron MY, et al. Constraints on neural redundancy. Elife. 2018; 7:e36774.
- 455 **Holtmaat A**, Svoboda K. Experience-dependent structural synaptic plasticity in the mammalian brain. Nature Reviews Neuroscience. 2009; 10(9):647.
- 457 **Holtmaat AJ**, Trachtenberg JT, Wilbrecht L, Shepherd GM, Zhang X, Knott GW, Svoboda K. Transient and persistent dendritic spines in the neocortex *in vivo*. Neuron. 2005; 45(2):279–291.
- 459 **Huber D**, Gutnisky D, Peron S, O'connor D, Wiegert J, Tian L, Oertner T, Looger L, Svoboda K. Multiple dynamic representations in the motor cortex during sensorimotor learning. Nature. 2012; 484(7395):473.
- 461 **Kappel D**, Legenstein R, Habenschuss S, Hsieh M, Maass W. A dynamic connectome supports the emergence of stable computational function of neural circuits through reward-based learning. eNeuro. 2018; 5(2):ENEURO-0301.
- 464 **Kaufman MT**, Churchland MM, Ryu SI, Shenoy KV. Cortical activity in the null space: permitting preparation without movement. Nature neuroscience. 2014; 17(3):440.
- 466 **Krumin M**, Lee JJ, Harris KD, Carandini M. Decision and navigation in mouse parietal cortex. Elife. 2018; 7:e42583.
- 468 **Levy SJ**, Kinsky NR, Mau W, Sullivan DW, Hasselmo ME. Hippocampal spatial memory representations in mice are heterogeneously stable. bioRxiv. 2019; p. 843037.
- 470 **Loewenstein Y**, Kuras A, Rumpel S. Multiplicative dynamics underlie the emergence of the log-normal distribution of spine sizes in the neocortex *in vivo*. Journal of Neuroscience. 2011; 31(26):9481–9488.

- 472 Loewenstein Y**, Yanover U, Rumpel S. Predicting the dynamics of network connectivity in the neocortex. *Journal
473 of Neuroscience*. 2015; 35(36):12535–12544.
- 474 Minderer M**, Brown KD, Harvey CD. The Spatial Structure of Neural Encoding in Mouse Posterior Cortex during
475 Navigation. *Neuron*. 2019; .
- 476 Minerbi A**, Kahana R, Goldfeld L, Kaufman M, Marom S, Ziv NE. Long-term relationships between synaptic
477 tenacity, synaptic remodeling, and network activity. *PLoS biology*. 2009; 7(6):e1000136.
- 478 Moczulska KE**, Tinter-Thiede J, Peter M, Ushakova L, Wernle T, Bathellier B, Rumpel S. Dynamics of dendritic
479 spines in the mouse auditory cortex during memory formation and memory recall. *Proceedings of the Na-
480 tional Academy of Sciences*. 2013; 110(45):18315–18320.
- 481 Montijn JS**, Meijer GT, Lansink CS, Pennartz CM. Population-level neural codes are robust to single-neuron
482 variability from a multidimensional coding perspective. *Cell reports*. 2016; 16(9):2486–2498.
- 483 Mulliken GH**, Musallam S, Andersen RA. Forward estimation of movement state in posterior parietal cortex.
484 *Proceedings of the National Academy of Sciences*. 2008; 105(24):8170–8177.
- 485 Ni A**, Ruff D, Alberts J, Symmonds J, Cohen M. Learning and attention reveal a general relationship between
486 population activity and behavior. *Science*. 2018; 359(6374):463–465.
- 487 Pedregosa F**, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, Blondel M, Prettenhofer P, Weiss R,
488 Dubourg V, et al. Scikit-learn: Machine learning in Python. *Journal of machine learning research*. 2011;
489 12(Oct):2825–2830.
- 490 Rokni U**, Richardson AG, Bizzi E, Seung HS. Motor learning with unstable neural representations. *Neuron*. 2007;
491 54(4):653–666.
- 492 Rule ME**, O’Leary T, Harvey CD. Causes and consequences of representational drift. *Current opinion in neuro-
493 biology (in press)*. 2019; .
- 494 Rumpel S**, Triesch J. The dynamic connectome. *e-Neuroforum*. 2016; 22(3):48–53.
- 495 Semedo JD**, Zandvakili A, Machens CK, Byron MY, Kohn A. Cortical areas interact through a communication
496 subspace. *Neuron*. 2019; 102(1):249–259.
- 497 Singh A**, Peyrache A, Humphries MD. Medial prefrontal cortex population activity is plastic irrespective of
498 learning. *Journal of Neuroscience*. 2019; 39(18):3470–3483.
- 499 Tonegawa S**, Pignatelli M, Roy DS, Ryan TJ. Memory engram storage and retrieval. *Current opinion in neurobi-
500 ology*. 2015; 35:101–109.
- 501 Wächter A**, Biegler LT. On the implementation of an interior-point filter line-search algorithm for large-scale
502 nonlinear programming. *Mathematical programming*. 2006; 106(1):25–57.
- 503 Widrow B**, Stearns SD. *Adaptive Signal Processing*. Prentice-Hall, Inc.; 1985.
- 504 Widrow B**, Hoff ME. Adaptive switching circuits. Stanford Univ Ca Stanford Electronics Labs; 1960.
- 505 Widrow B**, Hoff ME. Associative storage and retrieval of digital information in networks of adaptive ‘neurons’.
506 In: *Biological Prototypes and Synthetic Systems* Springer; 1962.p. 160–160.
- 507 Ziv Y**, Burns LD, Cocker ED, Hamel EO, Ghosh KK, Kitch LJ, El Gamal A, Schnitzer MJ. Long-term dynamics of CA1
508 hippocampal place codes. *Nature neuroscience*. 2013; 16(3):264.

509 **Supplemental Figures**

510 **Fig. 2-S1: Behavioral stability**

511 It is possible that changes in population codes relate to systematic changes in behavior over time.
 512 As described in Driscoll et al. (2017), these experiments were performed only after mice achieved
 513 asymptotic performance in speed and accuracy on the task. Nevertheless, details of behavior are
 514 important.

515 Across all mice and behavioral variables, there was a statistically-significant difference in means
 516 between 91% pairs of sessions ($p < 0.05$; Bonferroni multiple-comparison correction for a 0.05 false
 517 discovery rate (FDR)). However, the average effect size ($\Delta\mu/\sigma$, i.e. Cohen's d) was small, at 10–16%
 518 per animal. We could also partially predict the recording session based on 10-second kinematics
 519 trajectories (position, velocity, head-direction). Under cross-validation, kinematics could predict
 520 the recording session 9–17% above chance. We used a linear decoder to predict an indicator vector
 521 with a 1 in the session corresponding to the given kinematics trajectory, and 0 otherwise. The pre-
 522 dicted session was assessed as the session with the largest predicted value under cross-validation,
 523 and chance level assessed by permuting session identities.

524 Most of this predictive power came from differences in the forward movement in the initial
 525 portion of the T-maze. Much of this appeared to be daily variability, rather than drift (Fig. 2-S1).
 526 We found a small but significant systematic decrease in forward velocity in mouse 4 (Pearson cor-
 527 relation between recording day and median forward velocity of -0.9, two-tailed $p < 0.05$).

528 This suggests that each mouse exhibited small but detectable daily variability in their behavior.
 529 Most variability was unsystematic, and therefore unrelated to the slow changes in neural codes
 530 studied here. We expect changes in forward speed in mouse 4 to contribute to apparent drift in
 531 some cells. However, the results presented here generalize across mice 1, 3, and 5, which exhibited
 532 stable behavior.

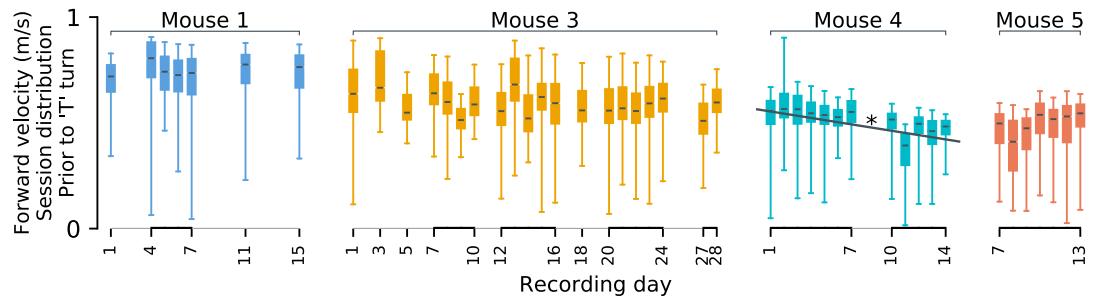


Figure 2—figure supplement 1: Statistics of forward motion show small daily variations. Each mouse's velocity in the initial (forward) segment of the 'T' maze varies slightly between days. Across days, behavior is broadly similar. Differences in means (black lines), although minuscule, are often statistically significant. Systematic drift-like trends appear absent from mice 1 and 3. A statistically significant trend is present for mouse 4 ($p < 0.05$). We show only forward velocity here, as other kinematics variables exhibited less variability.

⁵³³ **Figure 4-S1, concatenated decoder performance depends on the rank of the drift**

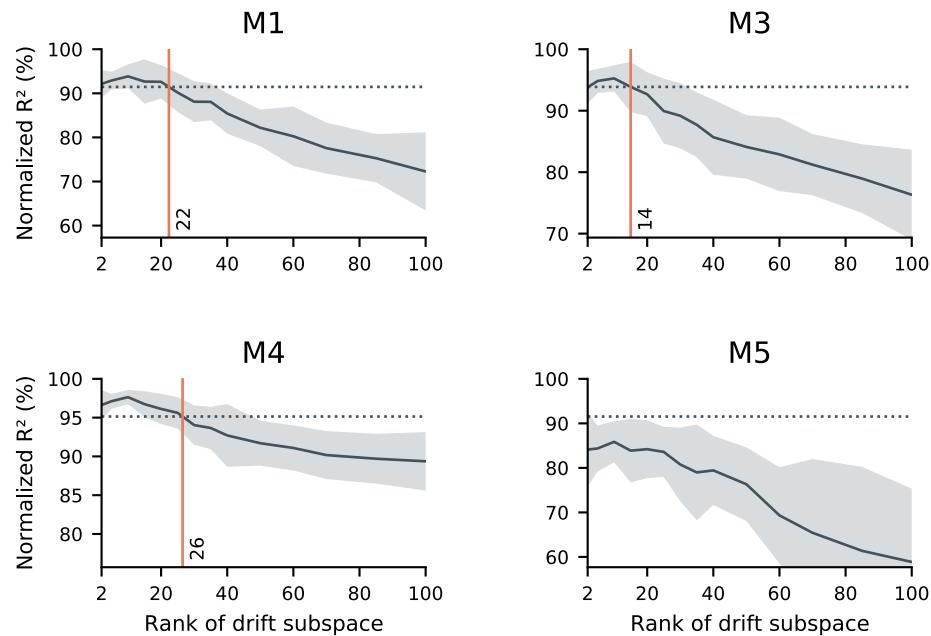


Figure 4—figure supplement 1: sufficiently low-rank drift resembles the data in terms of the performance of a concatenated decoder. Here, we further explore the null model introduced in Figure 3d. As in Figure 3d, we simulated random drift in the neural readout. We matched the null model to the statistics of neural activity, the within-day decoding accuracy, and the performance degradation when generalizing between days. In these simulations, we explore the scenario that the drift may be confined to a (randomly-selected) low-dimensional subspace. We evaluated a range of dimensionalities for the drift subspace (horizontal axes), and evaluated the performance of a concatenated decoder on simulated data. While unconstrained drift prevents the identification of a concatenated decoder with good performance (Fig. 3d), sufficiently constrained drift does not. In these simulations, we found that constraining drift to a subspace of rank 14–26 (red vertical lines) led to similar performance as the data (dashed horizontal lines) in all subjects except for mouse 5. We speculate that this is because Mouse 5 had limited data and poor generalization of single-day decoders over time, but other scenarios are possible. Black traces reflect the mean over 20 random simulations, and shaded regions reflect one standard deviation.

534 Figure 5-S1, online learning with LMS: additional subjects

535 We present an example of the LMS decoding algorithm on mouse 4 in the main text. We show
536 similar results here for mice 1, 3, and 5. We observed inter-day weight changes of 7.6-10.4%, con-
537 sistent with observed rates of change in the volume of dendritic spines in other studies. We used a
538 learning rate of 4×10^{-4} 1/sample, which led to error rates within 5 – 15% of the concatenated decoder
 (depending on the random selection of training and testing trials used for validation)

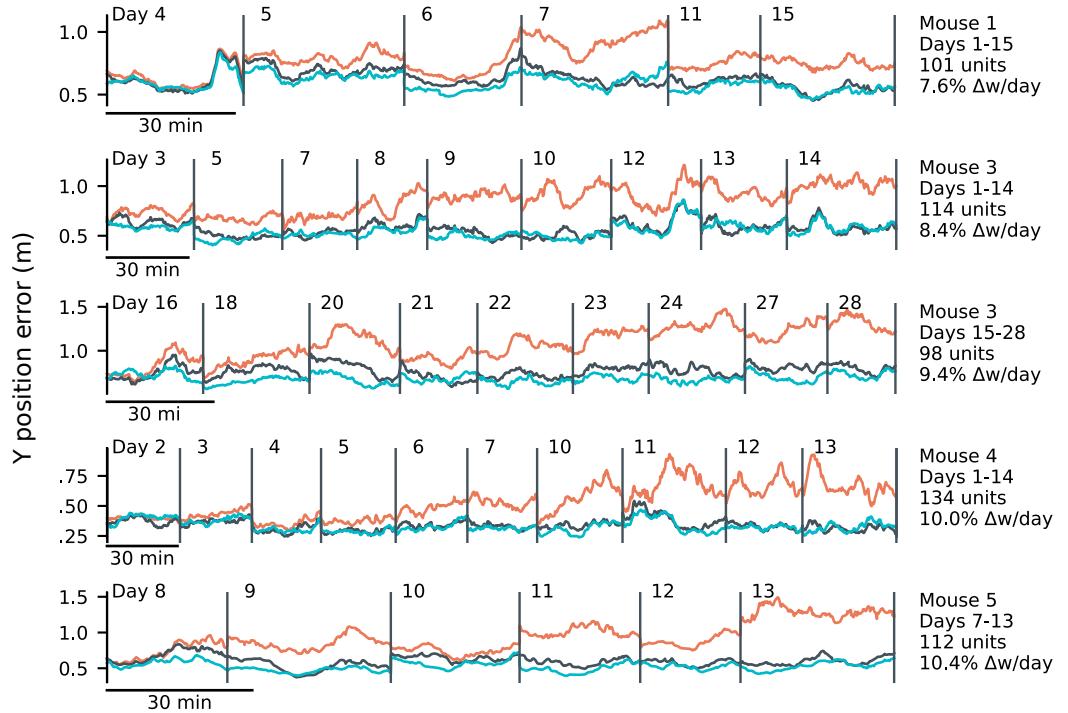


Figure 5—figure supplement 1; LMS results for mice 1, 3, 4, and 5. Results of applying the online LMS algorithm with a learning rate of 4×10^{-4} /sample. Errors reflect the mean absolute error over ten minute intervals. LMS (black) achieves errors comparable to an offline decoder trained on all sessions ("concatenated", blue), and outperforms a fixed decoder trained on the initial day (red). Only times within a trial were used for training. We present two spans of time from Mouse 3, reflecting two largely non-overlapping populations of tracked neurons on non-overlapping spans of days.

539

⁵⁴⁰ **Figure 5-S2: The plasticity level required to track drift varies with population size**

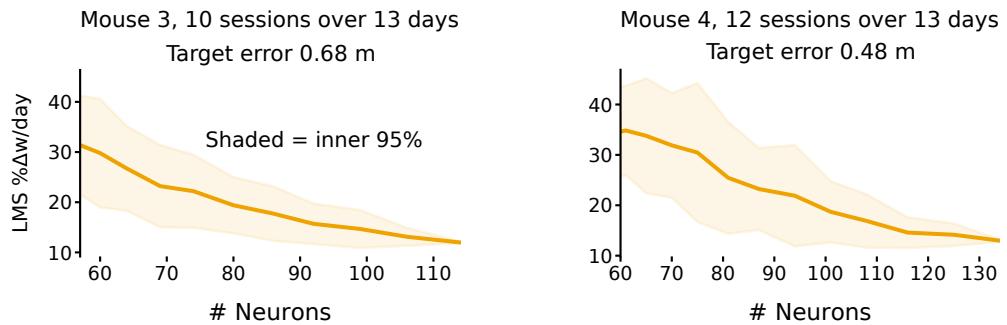


Figure 5—figure supplement 2; Smaller populations require more plasticity to achieve target error levels. These plots show the daily weight changes required to track drift when decoding forward positions as a function of population size for mice 3 and 4. Smaller populations require more plasticity. The target error (M3: 0.68 m, M4: 0.48 m) was set based on the performance of LMS on the full population (M3: 114 neurons, M4: 134 neurons). For each sub-population size, 50 random sub-populations were drawn, and the learning rate was optimized to achieve the target error level. Shaded regions reflect the inner 95th percentile over all sampled sub-populations. Weight change was assessed as the weight change between the end of consecutive sessions and normalized by the overall average weight magnitude.

541 Figure 5-S3: Extrapolation to larger populations

542 We saw in Figure 5-S2 that the amount of synaptic plasticity required to track a given error de-
 543 creases for larger populations. In this study, we examine recorded populations of ~100 neurons.
 544 Typically, the number of inputs to a given neurons is much larger than this, on the order of thou-
 545 sands. The ~10% weight change per day reported by LMS could therefore be an over-estimate of
 546 the amount of plasticity required. To address this, we extend the LMS analysis to larger popula-
 547 tions by combining neurons from different mice. This procedure destroys population correlations,
 548 and requires aligning activity from different mice. Despite this, the pooled populations yields a
 549 useful study of how plasticity scales with population size.

550 To align data from different mice, we matched trials based on the current and previous trial
 551 cue, and converted the neural time-series into location-based pseudotime, representing the frac-
 552 tion of the maze completed between 0 and 100%. This allowed us to register neuronal signals from
 553 different trials on different mice. We constructed a synthetic population of 1238 cells, covering a
 554 six-session-long recording period. We allowed up to two-day recording gaps between consecutive
 555 sessions from the same mouse. We found that larger populations could achieve the same perfor-
 556 mance as ~100 cells with a ~4% weight change per day.

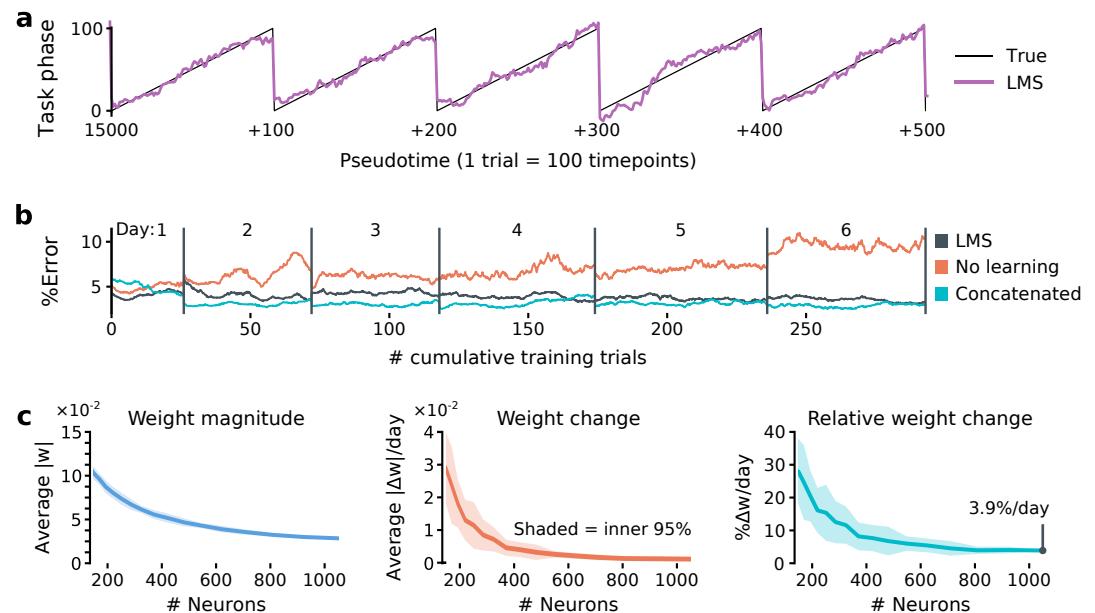


Figure 5—figure supplement 3; The plasticity required to achieve a fixed error level decreases for larger populations. (a) Trial pseudotime (% of trial complete; black) can be decoded from a synthetic pooled population (1238 cells) using the LMS algorithm (violet: prediction). **(b)** Similarly to the single-subject results, LMS tracks changes in the population code over time. In this case, a learning rate of $8 \times 10^{-4}/\text{sample}$ achieved comparable error to a concatenated decoder. The larger population permits better decoding error of ~5%, compared to the ~15–20% error in forward position decoded from ~100 neurons. **(c)** As population size increases, both the weight magnitudes (left) and the rates of weight change (middle) decrease. Small populations could not achieve the error rates possible using the full population, even with very large learning rates. We therefore set the target error a bit higher, at 13% chance level. This is comparable to the error rates seen in individual mice using ~100 cells. Overall, the required percentage weight change decreased for larger populations (right).